Bayesian models for Large-scale Hierarchical Classification

Siddharth Gopal*, Yiming Yang*, Bing Bai†, Alexandru Niculescu-Mizil†
*Carnegie Mellon University †NEC Labs, Princeton

Motivation

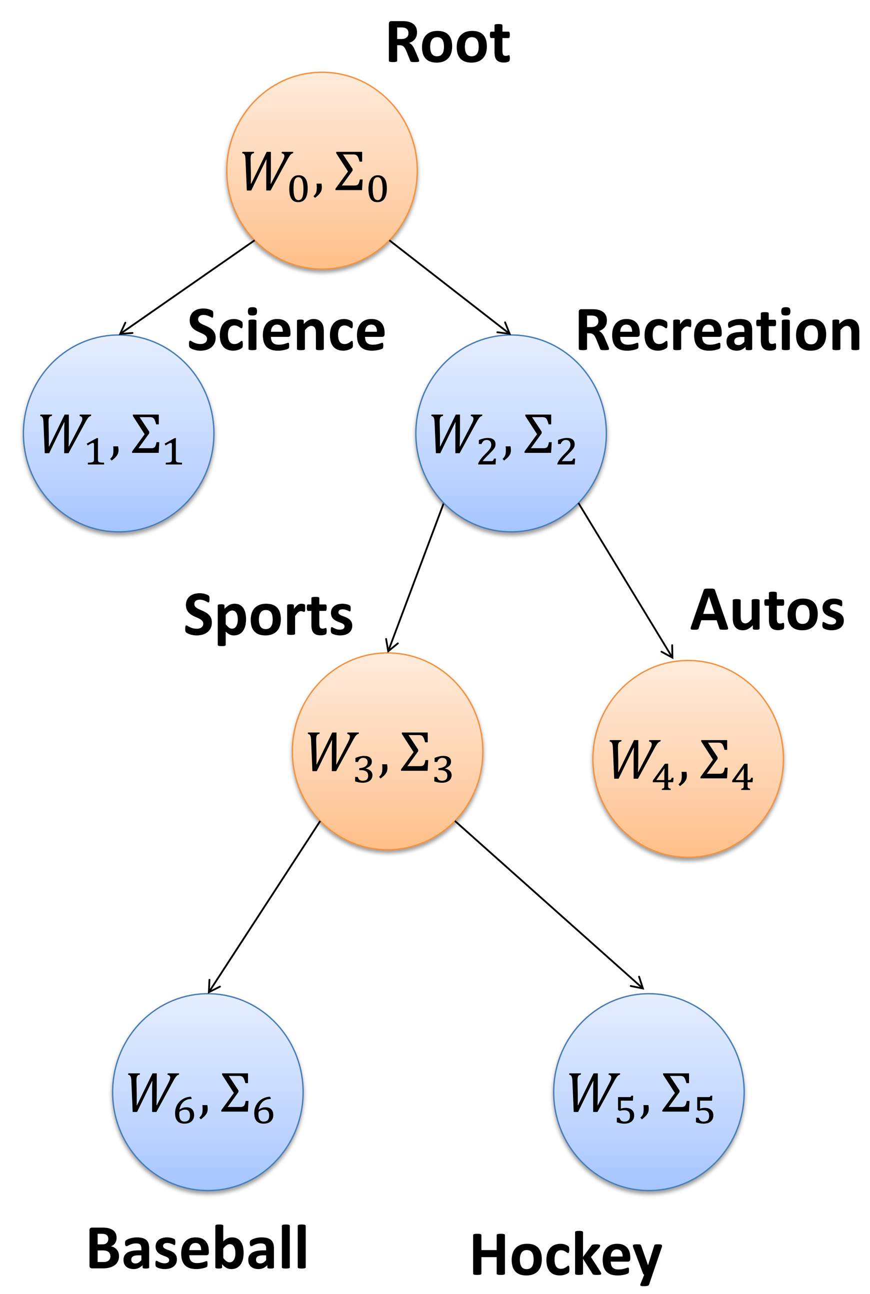
Hierarchies are ubiquitous – Yahoo! Directory, Open Directory Project for webpages, International patent taxonomy for patents etc. How to classify incoming data into an existing hierarchy. Specifically,

1. How to leverage the hierarchical dependencies between class-labels to improve classification ?
2. How to do it in a scalable manner for hierarchies with thousands of classes ?

Hierarchical Bayesian Modeling

A hierarchical Bayesian model where the prior distribution for the parameters at a node is a Gaussian centered at the parameters of its parent node. Given training data $D = \{x_i, t_i\}_{i=1}^{i=N}$, a parent function π, nodes Y,

$$W_y \mid W_{\pi(y)}, \Sigma_{\pi(y)} \sim N(W_{\pi(y)}, \Sigma_{\pi(y)})$$
$$t_i \mid x_i \sim Multinomial(softmax(\mathbf{W}, x))$$



Modeling the covariance structures gives different ways of sharing information in the hierarchy.

MODEL M1 : Node-specific covariance parameter.

$$\Sigma_y^{-1} = \alpha_y \mathbf{I}$$
$$\alpha_y \sim \Gamma(a_y, b_y) \quad \forall\, y$$

MODEL M2 : Feature-specific covariance. Sub-topics *baseball* and *Hockey* might be similar along features like *'players', 'Game'* but dissimilar along *`puck', `pitch'* etc.

$$\Sigma_y^{-1} = diag(\alpha_y^1, \alpha_y^2, .., \alpha_y^d)$$
$$\alpha_y^{(i)} \sim \Gamma\left(a_y^{(i)}, b_y^{(i)}\right) \quad \forall\, y$$

MODEL M3 : Learns how the individual children nodes differ from the parent node. For e.g under a topic *`mammals', the subtopic 'whales'* is very distinct the other typical subtopics like *`carnivores', 'herbivores'.*

$$W_y \mid W_{\pi(y)}, \Sigma_y \sim N(W_{\pi(y)}, \Sigma_y)$$
$$\Sigma_y^{-1} = \alpha_y \mathbf{I}$$
$$\alpha_y \sim \Gamma(a_y, b_y) \quad \forall\, y$$

Scalable Learning

A. Variational Inference: Computationally intensive due to matrix inversions. Applicable for *small-scale data* with hundreds of features and classes.

B. Partial MAP Inference: The computationally intensive part can be substituted with an MLE estimation followed by a MAP approximation for the posterior.

$$argmax_{\mathbf{W}} E[\log P(\mathbf{D}|\mathbf{W}, \boldsymbol{\alpha}) P(\mathbf{W}, \boldsymbol{\alpha})]$$

Applicable for large-scale data with several *hundreds of classes and tens of thousands of features*.

C. Parallel Partial MAP Inference: By replacing the soft-max function with multiple binary logistic functions, the MLE estimation can be parallelized by optimizing the parameters at odd (red) and even (blue) levels in parallel. Applicable to very large-scale data with tens of thousands of classes and millions of features.

Setting Prior Parameters

A **data dependent** way to set prior parameter based on asymptotic covariance of the MLE estimator i.e. **Fisher Information** matrix. For class-label y, the Fisher Information is given by,

$$I(y) = \sum p(y|x)(1 - p(y|x)) x x^T$$

Set the priors a_y, b_y to be the observed $I(y)^{-1}$ from the data. For example, for Model M2, $a_y, = 1,\ b_y = I(y)^{-1}$.

Results